

PROGRAMMING KEYBOARD status is a mixed bag. Bally still seems to have a July-August date for the appearance of the Keyboard. But there is an internal question now going on at Bally that asks if it might not be better to have a \$300. Keyboard with lesser capability (but expandable). The marketing surveys they have been running have indicated some kind of resistance (understandable) to a \$600+ unit. They have worked up a piece of hardware as a mockup to use internally for evaluation. But the decision (300,600,maybe both?) must come soon if a 12-16 week production span is needed after go-ahead to stay within the 3rd Quarter window they had set up last year for availability.

HACKER'S MANUAL has appeared. I understand that it is being distributed by some dealers as an addendum to the regular manual. If your dealer does not have a copy of this 18-page document for you, I can make a copy and ship it out for \$2. Most of the data has already been included in the various issues of the ARCADIAN as our fellow subscribers have discovered them on their own. The 'new' material includes: some words on the I/O ports, a few words on the light pen interface, a block diagram of the sound synthesizer and description, and considerable detail as to wiring changes in the cassette interface to allow the addition of a printer jack.

LATE DATA on product availability...

2005	Star Battle	19.95	out in February
2007	Pinball	24.95	} March
3003	Grand Prix/Demolition Derby	19.95	
3004	Desert Fox/ Drag Race	19.95	
4004	Music	24.95	
5003	Backgammon/Checkers	19.95	

INTERACTIVE PROGRAMMING is being worked on by Jim Unroe. This is a scheme by which two machines can talk to each other via the cassette interfaces.

INTERCONNECTION to the S.D. Sales Z-80 CPU BOARD(kit \$139.,P.O.Box 28810 Dallas 75228) is being explored by Pete Wishart up in Canada. He has developed a wiring scheme to to into the 50-pin connector on the back of the Arcade and wind up with an S-100-compatible interface. Still some bugs to be worked out.

DEALER in the Arkansas area is J.W.Taylor, 611 North 2nd, Cabot,72023 who has an extensive supply, and I believe sends items postpaid.

LETTERHEAD of this issue was donated by Herb Weintraub. It is an interesting idea...

MENU can be called up by the following, donated by Martin Nason:

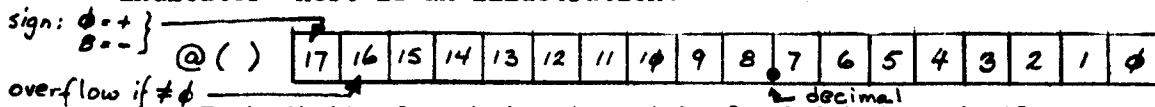
```
10 INPUT K
20 CALL K          insert 3174
```

The menu will appear, and function fully(don't use the BASIC overlay card) but why does it not work if you just CALL 3174?

arcadian

ONBOARD CALCULATOR was very briefly mentioned in January. Here is some data on this feature. With this routine, it is possible to perform the four arithmetic functions with decimals, and use numbers much bigger than the Tiny BASIC limitation of 32767. But it takes up a lot of space. The operation is listed as \$ N @(A), @(B), @(C) where N is the desired function + - ÷ *

A is an input address, B is an input address, and C is the answer address. Each address is the beginning location of an 18-consecutive string, so that we could have A extending from 0 to 17, 18 to 35, 36 to 53, etc. B and C are similar. Within each of these sets, the decimal point is located at the near-center, the sign of the number is at the end, adjacent to an overflow indicator. Here is an illustration:



Each digit of each input must be loaded independently, as well as its sign.

As an example, let us multiply 374.2913 by 96.7 to get 36193.96871:
 Note the location of the decimal point and work from there-

Load the first input:

10 @(10)=3; @(9)=7; @(8)=4; @(7)=2; @(6)=9; @(5)=1; @(4)=3

Load the second input

20 @(27)=9; @(26)=6; @(25)=7

The registers will look like this:

@	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	ϕ
	ϕ	ϕ	ϕ	ϕ	ϕ	ϕ	ϕ	3	7	4	.	2	9	1	3	ϕ	ϕ	ϕ
@	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
	ϕ	ϕ	ϕ	ϕ	ϕ	ϕ	ϕ	ϕ	9	6	.	7	ϕ	ϕ	ϕ	ϕ	ϕ	ϕ

List the operation:

30 \$*@(0), @(18), @(36) ;RUN

The answer register looks like this:

@	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36
	ϕ	ϕ	ϕ	ϕ	ϕ	3	6	1	9	3	.	9	6	8	7	1	ϕ	ϕ

and to recover it, include

```
40 FOR A = 53 TO 36 STEP -1
50 TV=@(A)
60 NEXT A
```

Which will yield 000003619396871000

This technique will suppress the leading zeros - adjust the values in lines 70 and 80 to fit the location of your answer address:

```
60 Z=1
70 IF @(53)="8" PRINT "-" . If answer is a negative
80 FOR B=52 TO 44 STEP -1
90 IF @(B)="0" IF Z GOTO 120
100 Z=0
110 TV=@(B)
120 NEXT B
```

The locations A,B,and C can be mixed up, or set equal to each other, or use other locations for memory, saving them for later use.

arcadian

FORMATTING (PRINT #N) The following is a contribution from Tom Wood with some of my added comments and example. " A PRINT statement containing a #N value is most interesting. Apparently the value for N following the # sets the size of a 'window' to be left on the screen for each variable in the statement. Variables will be printed right-justified within that window.

A = 34; B = 973; C = -88; PRINT #4,A,B,".",C yields the following -

			3	4			9	7	3	.		-	8	8
--	--	--	---	---	--	--	---	---	---	---	--	---	---	---

We created a window of 4 character spaces wide for each variable on the PRINT line, noting that . is not a variable. The window is effective for the entire PRINT line, or until there is another #N " - Wood.

This gives you the capability to create tabulated columns across the screen. To get this:

JOHN DOE 391-7721
BILL WILSON 271-8143
HARRY JONES 814-6392

Program this:

```
10 A=391;B=7721;C=271;D=8143;  
E=814;F=6392  
20 PRINT" JOHN DOE",#6,A,"-",#4,B  
30 PRINT"BILL WILSON",#6,C,"-",#4,D  
40 PRINT"HARRY JONES",#6,E,"-",#4,F
```

With the Onboard Calculator routine giving decimal calculation, you can start setting up material necessary for payroll accounting with answers in nice neat columns. Has anyone done any business programs?

PROGRAMS HERE, contributed by subscribers, include such games as CHECKERS, STRATEGY FOOTBALL, SLOT MACHINE, BALLY TREK, etc., and which are quite lengthy. I really haven't had time to give them a good scrubbing, but plan on doing so next month, and have them available for subscribers. I finally received a box of C-10 tapes from Microsystems, so now I can get organized.

PROGRAMS DIRECT from subscribers:

o Bob Weber 6594 Swartout Rd. Algonac MI 48001 has the following available at \$2 plus a C-30 tape:

SUB SEARCH	ALIEN PATROL	CALENDAR
SLOT MACHINE	CONCENTRATION	TIC TAC TOE
FLIGHT SIMULATOR	HANGMAN	MATH QUIZ
OTHELLO	MASTERMIND	SPACE CHASE

o Ron Schwenk 6988 Lincoln Creek Circle, Carmichael CA 95608

MASTERMIND	ONE CHECK	BATNUM
------------	-----------	--------

o Bob Strand 10665 E. Foix Ave. Norwalk CA 90650 \$7 for the lot...

STAR BATTLES	4 DIGIT GUESS	REMEMBER
ANGLE GAME	SLOT MACHINE	NUMBER WAR

LUNAR LANDER(enhanced/expanded)

o George Hale P.O.Box 186 Lee's Summit MO 64063 has a shoot-it-down type of game for two that he calls SONIC SATELLITE. This will be available as a listing for \$4., as a cassette tape @ \$8.50, or loaded on your tape @ \$6.50. George will be selling Bally-oriented goods through Applications Programming Enterprise.

FOR SALE Bally ARCADE with BASIC, CLOWNS, and BASEBALL, \$275. W.KIM, 776 Via Catalina, Burbank, CA 91504 (213-767-3963)

arcadian

<u>MEMORY MAP</u>	Decimal	Hexadecimal
On Board ROM	0- 8191	0-1FFF
Bally BASIC ROM	8192-12287	2000-2FFF
Screen Memory Area	16384-20479	4000-4FFF
Bally BASIC Graphics/Program Area	16384-19983	4000-4E10
Bally BASIC Scratchpad Memory Area	20000-20463	4E20-4FEF
Tape Input Buffer	20002-20049	4E22-4E51
Variables begin at	20078	4E6E
Line Input Buffer (104 characters)	20180-20283	4ED4-4F3B
Stack Area	20284-20462	4F3C-4FEE
Text Area	24576-22777	A000-A707
Note Lookup Table	12046	2FOE for CR(13 _o)

The above was extracted from the Hacker's Manual.

SPACE SAVER has been located by Bob Weber - If a PRINT "X" is not followed by another command, the final " is not needed. "A byte saved is a byte available for another statement."

ANOTHER DIVISION ROUTINE that prints a decimal answer has been developed by Pete Bowman, This one is a bit laborious as you have to enter a @() for each decimal wanted, in line 80.

```
10 PRINT "X- Y = Z"      50 FOR W = 1 TO N (where N is the number of
20 INPUT "X=?" X          decimal digits desired)
30 INPUT "Y=?" Y          60 @(W) = (RM*10) ÷ Y
40 Q = X ÷ Y              70 NEXT W
                        80 PRINT "Z=", #1, Q, " .", @(1), @(2), @(3), ...@(N)
```

NOTE TIME has been noted by many to control speed of operations to some extent. Setting it =0 makes things operate the fastest. Negative numbers yield very slow results. You can also go back and forth to tape faster with :PRINT;NT=1;LIST Using NT=0 here doesn't always work.

PROGRAMS INCLUDED this month are short enough to put on a page. The form that I used was provided by Chuck Thomka, 1228 W.222 St. Torrance CA 90502.

It is a handy way to keep things in order. Program listing should be reviewed as a training aid, to help in your own understanding.

PROGRAM NAME

```

Line #      Statement(s)
1 .SIMON
2 .BY BRETT BILBRAY
3 .AND JOE BORRELLI
10 CLEAR; &(0)=7; &(1)=7; &(2)=0
; &(3)=0; &(9)=30; NT=0; CX=47
; CY=-20; PRINT "SCORE:";
NT=30
20 BOX 0,20,22,22,1; BOX 0,20,
20,20,2
30 BOX 0,-20,22,22,1; BOX 0,
-20,20,20,2
40 BOX 20,0,22,22,1; BOX 20,0,
20,20,2
50 BOX -20,0,22,22,1; BOX -20,0,
20,20,2
60 A=A+1; @ (A)=RN(4)
70 FOR X=1 TO A; IF @ (X)=1
GOSUB 130; GOTO 110
80 IF @ (X)=2 GOSUB 200;
GOTO 110
90 IF @ (X)=3 GOSUB 300;
GOTO 110
100 GOSUB 400; GOTO 110
110 NEXT X; FC=0; FOR X=1 TO A
120 B=JX(1); C=JY(1); IF B=0 IF
C=0 GOTO 120
130 IF B#0 D=(1+B)/2+3
140 IF C#0 D=(1-C)/2+1
150 IF D#@ (X) NT=55; MU=33; MU=48
; MU=48; NT=3; FC=0; FOR X=1 TO
5000; NEXT X; GOTO 10
160 GOSUB D*1000; GOTO 170
170 FC=0; NEXT X
180 CX=23; CY=-30; NT=0; PRINT A;
NT=30
190 FOR O=1 TO 500; NEXT O;
GOTO 60
1000 FC=44; BOX 0,20,20,20,1; MU=
49; FOR O=1 TO 255-2*A; NEXT O
; BOX 0,20,20,20,2; RETURN
2000 FC=204; BOX 0,-20,20,20,1;
MU=51; FOR O=1 TO 255-2*A;
NEXT O; BOX 0,-20,20,20,2;
RETURN
3000 FC=252; BOX -20,0,20,20,1; MU
=53; FOR O=1 TO 255-2*A; NEXT
O; BOX -20,0,20,20,2; RETURN
4000 FC=134; BOX 20,0,20,20,1; MU=
55; FOR O=1 TO 255-2*A; NEXT O
; BOX 20,0,20,20,2; RETURN

```

(Changes copied from page 45)

SIMON CORRECTIONS: Make the following changes in the program:

Revise line 10 CLEAR; &(0)=7; &(1)=7; &(2)=0; &(3)=0; &(9)=30;
 NT=0; CX=47; CY=20; PRINT "SIMON"; B=7; A=0; CX=47;
 CY=-20; PRINT "SCORE:"; NT=5

Delete lines 70,80,90,100,160,170

Add lines 70 FOR X=1 TO A

80 GOSUB @ (X)x1000

155 IF D=1 GOSUB 1000; GOTO 170

160 IF D=2 GOSUB 2000; GOTO 170

164 IF D=3 GOSUB 3000; GOTO 170

166 GOSUB 4000

Revise line 150 IF D#@ (X) NT=55; MU=33; MU=48; MU=48; NT=3;

FC=0; GOTO 10

In lines 1000,2000,3000,4000 delete the -2xA after 1 TO 255

Line # 5 .CLDCK BY J. COUSINS

```

6 .
7 .
10 CLEAR; BC=0; FC=22
20 INPUT "HOURS" H
30 IF H>23 GOTO 20
40 INPUT "MINUTES" M
50 IF M>59 GOTO 40
60 INPUT "SECONDS" S
70 IF S>59 GOTO 60
80 CLEAR
90 FOR I=1 TO 70
100 NEXT I; NT=0
110 B=B+1
120 IF B<=59 GOTO 180
130 B=0; M=M+1
140 IF M<=59 GOTO 180
150 M=0; H=H+1
160 IF H<=23 GOTO 180
170 H=0; M=0; B=0
180 CX=-12; CY=20; A=15
190 IF H<13 A=0
200 A=A+65
210 TV=A; TV=77
220 CX=-30; CY=0; T=H
230 GOSUB 290
240 TV=58; T=M
250 GOSUB 290
260 TV=58; T=8
270 GOSUB 290
280 NT=1; MU=76; GOTO 90
290 A=ABS(T-I0); B=R.M
300 TV=A+48; TV=B+48
310 RETURN

```

USE OF SHADED AREA IS FOR 2ND OR MORE LINES OF MULTI-LINE STATEMENTS

DO NOT ENTER A SPACE BETWEEN LINE # AND STATEMENT THIS IS DONE BY THE UNIT

Line # 5 .CONVERT HEX TO DECIMAL

```

6 . BY ERNIE SAMS 3-1-79
7 .
10 CLEAR
20 M=0; PRINT
30 FOR N=0 TO 3
40 PRINT "HEX #"
50 K=KP
52 IF K<47 GOTO 200
54 IF K>70 GOTO 240
56 IF K>57 IF K<65 GOTO 200
60 TV=K; PRINT
70 N=K-48
80 IF K>57 N=N-7
90 IF A=0 IF N>7 M=-32767;
M=M+(4095*(N-7))+N-7;
M=M-4096; IF N>8 M=M-1;
GOTO 150
100 IF A=0 IF N=8 GOTO 150
110 IF A=0 M=M+(4096*N)
120 IF A=1 M=M+(256*N)
130 IF A=2 M=M+(16*N)
140 IF A=3 M=M+(1*N)
150 NEXT A
160 PRINT; PRINT "THE DECIM
AL # IS: "; M
170 GOTO 20
200 PRINT; TV=K; PRINT "IS A
N INVALID HEX #"; GOTO 40

```

USE OF SHADED AREA IS FOR 2ND OR MORE LINES OF MULTI-LINE STATEMENTS

DO NOT ENTER A SPACE BETWEEN LINE # AND STATEMENT THIS IS DONE BY THE UNIT

command name	function		
box	draws a rectangle on the screen & has options for building picture prototype lists	select	causes picture prototypes to be switched round-robin fashion on the screen
*change	changes the values of an endpoint in a picture prototype list	snap	takes a screen image in rectangular bounds and makes it into a movable picture prototype
circle	draws an ellipse on the screen & has options for building picture prototype lists	sync	tells the system how much time to devote to interrupt-level updating versus command processing
clear	clears the screen	*vip	allows a macro to be executed at interrupt level (stands for "very important program")
*close	closes off an open picture prototype list		
colors	chooses 4 colors of 256 for screen use		
*compile	compiles code for speed		
copy	makes a copy of a picture prototype with a new name		
delete	deletes and reclaims storage of a named thing		
display	causes a picture prototype to be exclusive or'ed onto the screen and be updated when necessary		
*film	sets up filming mode for a Super 8 camera		
*fetch	retrieves a given endpoint in a picture prototype list		
get	gets a macro, array, picture prototype list, etc. from tape, disk, etc.		
group	collects picture prototypes into a group which can be referenced with a single name. Transformations may be done to the group as a whole or to individual members.		
help	prints commands and required argument types		
ieee	provides interface to IEEE bus		
input	used to input numbers, strings from terminal or passed argument lists		
line	draws a vector & has options for building picture prototype lists		
memory	gives a usage map of memory		
move	attaches a picture prototype to two variables, devices, etc. so that when they change, the prototype is automatically erased and redrawn in the new position with options for "exclusive or" or "load/store" read and write to screen		
*onerror	traps errors to a user's routines		
*open	allocates storage and starts up a picture prototype list		
*pattern	allows a pixel list to be directly built rather than snapped		
play	interprets a string, array or picture prototype as a musical score to be played by the three-voice synthesizer		
put	stores a macro, array, picture prototype list, etc. on tape, disk, etc.		
rename	renames a named thing to a new name		
*rotate	like move but the prototype is rotated		
*scale	like move but the prototype is scaled		

ZGRASS COMMANDS are listed here. These are some of the unique ones planned for the Keyboard's language. The machine I saw had a total of 66 commands. This page followed "page 36" of the article reproduced in ARCADIAN #2.

Line # Statements(s)

```

1 . REVERSE
2 . BY BRETT BILBRAY
3 . AND MIKE TOTM
10 N=9; CLEAR
20 @ (I)=RND(9)
30 FOR K=2 TO N
40 @ (K)=RND(9)
50 FOR J=1 TO K-1
60 IF @ (K)=@ (J) GOTO 40
70 NEXT J; NEXT K
80 CY=25
90 PRINT "THE LIST IS"
100 T=0
110 GOSUB 200
120 CY=-20; CX=0; Q=0; NT=0
130 Q=KN(1)÷32+6; IF TR(1)
    =1 GOTO 160
140 IF 0#Q CX=-44; PRINT 0; Q=0
150 GOTO 130
160 T=T+1; NT=3
170 FOR K=1 TO Q÷2
180 Z=@ (K)
190 @ (K)=@ (Q-K+1)
200 @ (Q-K+1)=Z
210 NEXT K
220 GOSUB 200
230 FOR K=1 TO 9; IF @ (K)≠K
    GOTO 120
240 NEXT K
250 CX=-70
260 CY=-20; PRINT " YOU WON IN"
    ; TV=T÷10+48; TV=T÷10+48;
    PRINT "MOVES"
270 FOR X=1 TO 50; NEXT X;
    GOTO 10
280 CX=-5; CY=0
290 FOR B=1 TO N; TV=48+@ (B);
    TV=32; NEXT B
300 RETURN

```

GAME INSTRUCTIONS These games were sent by Brett Bilbray who welcomes comments and suggestions at 14430 Barclay, Dearborn, MI 48126.

SIMON: One player, Hand Controller
The computer shows you a pattern that you have to repeat, using joy stick controls.

REVERSE: One player, Hand Controller
The object is to get 9 number in order (smallest at the left) that are initially in random order. Use the knob to identify the numbers to be moved, and the trigger to move them.

REVERSE CORRECTIONS (from page 46)
Revise line 260 CY=-20; PRINT " YOU WON IN", ; TV=T÷10+48;
TV=T-T÷10x10+48; PRINT "MOVES"
270 GOTO10
280 CX=-50; CY=0

ARCADIAN

Robert Fabris, Boss
3626 Morrie Dr.
San Jose, CA 95127

FIRST CLASS